



A computer-based simulation of childbirth using the partial Dirichlet–Neumann contact method with total Lagrangian explicit dynamics on the GPU

Rudy Lapeer¹ · Zelimkhan Gerikhanov¹ · Said-Magomed Sadulaev¹ · Vilius Audinis¹ · Roger Rowland¹ · Kenda Crozier² · Edward Morris³

Received: 5 July 2018 / Accepted: 15 December 2018
© The Author(s) 2019

Abstract

During physiological or ‘natural’ childbirth, the fetal head follows a distinct motion pattern—often referred to as the cardinal movements or ‘mechanisms’ of childbirth—due to the biomechanical interaction between the fetus and maternal pelvic anatomy. The research presented in this paper introduces a virtual reality-based simulation of physiological childbirth. The underpinning science is based on two numerical algorithms including the total Lagrangian explicit dynamics method to calculate soft tissue deformation and the partial Dirichlet–Neumann contact method to calculate the mechanical contact interaction between the fetal head and maternal pelvic anatomy. The paper describes the underlying mathematics and algorithms of the solution and their combination into a computer-based implementation. The experimental section covers first a number of validation experiments on simple contact mechanical problems which is followed by the main experiment of running a virtual reality childbirth. Realistic mesh models of the fetus, bony pelvis and pelvic floor muscles were subjected to the intra-uterine expulsion forces which aim to propel the virtual fetus through the virtual birth canal. Following a series of simulations, taking variations in the shape and size of the geometric models into account, we consistently observed the cardinal movements in the simulator just as they happen in physiological childbirth. The results confirm the potential of the simulator as a predictive tool for problematic childbirths subject to patient-specific adaptations.

Keywords Finite element method · Biomechanics · Hyperelasticity · Obstetrics

1 Introduction

The biomechanical process of human childbirth involves intricate interactions between the two main agents, i.e. the fetus and the maternal abdominal and pelvic anatomy. More specifically, during the second stage of labour, the fetal head

comes into contact with the maternal bony pelvis and pelvic floor muscles due to the expulsive forces aiming to expel the fetus from the womb. From a purely physical perspective, this constitutes a mechanical contact problem which lies at the basis of various phenomena including:

- The ‘cardinal movements’ (CMs) of the fetal head which occur during physiological birth¹;
- The adverse effect on pelvic floor muscles when over-stretched and potentially resulting in incontinence following childbirth;
- The possibility of labour coming to a halt because the fetal shoulder impacts with the bony pelvis known as shoulder dystocia (SD);
- Unfavourable presentations such as brow and face.

Electronic supplementary material The online version of this article (<https://doi.org/10.1007/s10237-018-01109-x>) contains supplementary material, which is available to authorised users.

✉ Rudy Lapeer
r.lapeer@uea.ac.uk

¹ School of Computing Sciences, University of East Anglia, Norwich, UK

² School of Health Sciences, University of East Anglia, Norwich, UK

³ Department of Obstetrics and Gynaecology, Norfolk and Norwich University Hospital NHS Trust, Norwich, UK

¹ The critical cardinal movements of the fetal head in order of occurrence are: flexion, internal rotation, extension and external rotation.

In the longer run, the aim of the project is to create a patient-specific ‘virtual reality’ (VR) childbirth simulator capable of assessing the likelihood of normal and, more importantly, abnormal outcomes for individual cases prior to the actual event. This is of great clinical importance as it would allow clinicians to plan ahead, for example, to decide on an elective Caesarean Section (CS) if the simulation returns a high risk score on the occurrence of SD. Before such a sophisticated predictive simulator can be developed, the ‘normal’ interaction between fetus and maternal pelvic anatomy—nowadays referred to as physiological childbirth—has to be realistically modelled and simulated first.²

In this paper, we focus on the technical side of the R&D by describing the methodology we used to model the mechanical contact problem of physiological childbirth on computer. We used an explicit finite element (FE) formulation, better known as the Total Lagrangian Explicit Dynamics (TLED) method by Miller et al. (2007) to calculate the deformations of the soft tissues. The contact forces fed into TLED are calculated via a modified projection-based contact method (Yastrebov 2013). Validation is based on the quantitative and qualitative observations (or absence) of the CMs of normal childbirth. Even though the methodology is applied to modelling childbirth on computer in this paper, it can be easily adopted for other soft tissue simulations that include mechanical contact interactions, for example, surgical simulation and navigation, and robotic surgery.

2 Background

FE modelling of childbirth biomechanics related phenomena dates back to the late 1970s. McPherson and Kriewall (1980) modelled the deformation of fetal cranial bone when subjected to the uterine pressure during the first stage of labour using finite element Analysis (FEA). Due to limited processing power in those days, they created a FE model of the parietal bones only. Building on this preliminary model, Lapeer and Prager (2001) developed a 3D FE model of fetal head moulding and successfully validated it against a clinical study by Sorbe and Dahlgren (1983). Their work has been used in various applications including the effect of head moulding on the pelvic floor muscles (Silva et al. 2015), infant and baby head trauma (Loyd et al. 2015) and fetal growth (Libby et al. 2017).

The effect of labour, be it physiological or assisted, on the biomechanical behaviour of the pelvic floor muscles is another popular subject where FEA is used. Lien et al. (2004) predicted the stretching of the Levator Ani Muscle (LAM—the collection of pelvic floor muscles) using MR (magnetic resonance) images and FEA during the second stage of labour. Parente et al. (2009) assessed the effect of the variation in material properties of the pelvic floor muscles on their biomechanical behaviour during physiological childbirth. Silva et al. (2017) assessed the biomechanical properties of the pelvic floor muscles of continent and incontinent women, respectively, also using MRI data and (inverse) FEA. Oliveira et al. (2017) analysed the effect of episiotomy (incision made in the vagina to aid delivery or avoid damage to other tissues) on the pelvic floor muscles using FE models.

Last but not least, a number of attempts to model a VR childbirth simulator have been made since the early 1990s. Boissonnat and Geiger (1993) created a simple model of a fetal skull interacting with the bony pelvis. The model was not FE based, but contact forces and resulting moments on the head were based on the degree of volumetric interpenetration of the skull polyhedra into the pelvic polyhedra. Rotations of the head resembling the CMs were apparently observed during some simulations. Liu et al. (1996) proposed to improve Geiger’s model by developing a fully articulated fetal model and use FE analysis to determine the forces and moments acting on the fetal head though we are not aware of any further publications covering the suggested improvements. Buttin et al. (2009) modelled a fetus using two ellipsoids for the fetal head and trunk, respectively, and performed FEA on the descent of the fetus. They link their simulation to the BirthSIM system by Moreau et al. (2007) which is used for hands-on training and planning of obstetrics forceps delivery. The strengths of the reported research are the consideration of key soft tissues including the uterus and a mouldable (non-rigid) fetal head. These strengths are somehow weakened due to the simplification of the geometric models which are merely ellipsoids and no mention of the observation of the CMs or mechanisms of the fetal head. Gerikhanov et al. (2013) created a VR simulation with the aim of observing the cardinal movements starting with the least complex configuration of a rigid bony pelvis and a rigid fetal skull. The underlying model was based on rigid body mechanics. They observed flexion of the fetal head and internal rotation to some degree. They concluded that additional maternal anatomy such as the pelvic floor muscles would be required as a minimal configuration to observe the critical CMs (flexion, internal rotation, extension and external rotation). It is their proposal that has been further investigated in the research presented in this paper. The rigid body model by Gerikhanov et al. (2013) is bound to be inadequate due to the addition of soft tissues which significantly increase

² In the past, the term ‘normal’ childbirth was used when the baby is delivered vaginally without additional instrumental intervention. Following a debate amongst midwives and obstetricians on the ambiguity of ‘normal’, a birth without instrumental intervention is now called ‘physiological’ childbirth.

complexity; thus, a new model is required using a combination of explicit FE and mechanical contact methods.

3 Methodology

The proposed methodology is a combination of the Total Lagrangian Explicit Dynamics (TLED) explicit FE formulation which is coupled to a projection-based contact method to calculate the contact forces causing deformation in the soft tissues which subsequently cause the fetal head to rotate. The external body forces are the intra-uterine expulsion force and the maternal bearing down forces. The soft tissues (pelvic floor muscles, ligaments and uterine cervix) are modelled using tetrahedral elements and Neo-Hookean hyperelastic material properties. Other components in the model are static (maternal pelvis) or dynamic (fetus) rigid bodies. The fetus has one articulation, i.e. the fetal neck which couples the rigid fetal head to the rigid fetal body. It is comprised of a linear and torsional spring (see Table 3 for the stiffness coefficients) to resist translations and rotations, respectively. Bending (flexion/extension/lateral flexion) is constrained by the contact of the fetal head with the fetal body. Rotations in the transverse plane are constrained by the torsional spring. The remainder of this section describes each of the numerical TLED and contact methods separately and then consolidates these within the FE formulation. The implementation is covered at the end of the section.

3.1 TLED formulation

TLED (Miller et al. 2007) is a variation on the Lagrangian formulation of the finite element method (FEM). Belytschko et al. (2014) describe the difference between the updated Lagrangian (UL) and total Lagrangian (TL) formulations:

Dependent variable description:

UL—current or deformed configuration;
TL—reference or material configuration.

Derivatives of the dependent variables:

UL—evaluated with respect to the spatial coordinates;
TL—evaluated with respect to the material coordinates.

Strain measure:

UL—strain rate; TL—total strain.

The main advantage of the TL formulation is that a considerable number of variables can be pre-computed and then reused throughout the simulation, thus saving computation time.

We start by writing down the law of conservation of linear momentum (Belytschko et al. 2014):

$$\nabla_0 \cdot \mathbf{P} + \mathbf{b} - \rho_0 \ddot{\mathbf{u}} = 0 \quad (1)$$

where ∇_0 is the nabla operator in the reference (material) configuration, \mathbf{P} is the nominal stress, \mathbf{b} is a vector of body forces, ρ_0 is the density in the reference configuration and $\ddot{\mathbf{u}}$ is the acceleration vector or second derivative in time of the displacement vector \mathbf{u} .

Using the weak form (principle of virtual work) we multiply Eq. 1 with a variation δu and integrate over the reference domain Ω_0 :

$$\int_{\Omega_0} \delta u \frac{\partial \mathbf{P}}{\partial X} d\Omega_0 + \int_{\Omega_0} \delta u \mathbf{b} d\Omega_0 - \int_{\Omega_0} \delta u \rho_0 \ddot{\mathbf{u}} d\Omega_0 = 0 \quad (2)$$

where X is the position vector of a material point in the reference configuration and each of the three individual terms, respectively, represent:

- The internal energy δW^{int} ;
- The external energy $-\delta W^{\text{ext}}$;
- The kinetic energy δW^{kin} .

Considering the arbitrariness of the variation δu , the forces can be determined immediately from Eq. 2. The external forces f^{ext} are defined by the intra-uterine pressure and maternal bearing down forces (see Sect. 4.3 for actual values); the kinetic forces f^{kin} are determined in the TLED update loop (see Fig. 7).

This means that only the internal forces require further attention:

$$f^{\text{int}} = \int_{\Omega_0} \frac{\partial \mathbf{P}}{\partial X} d\Omega_0 \quad (3)$$

We approximate the internal forces f^{int} by nodal forces f by introducing shape functions N and the matrix of the shape function derivatives $\partial \mathbf{h}$. Expressing the first Piola–Kirchhoff (FPK) stress tensor \mathbf{P}^T in terms of the second Piola–Kirchhoff (SPK) stress \mathbf{S} whilst introducing the deformation gradient \mathbf{F} gives the nodal force:

$$f = \int_{\Omega_0} \frac{\partial N}{\partial X} \mathbf{P}^T d\Omega_0 = \int_{\Omega_0} \partial \mathbf{h} \mathbf{P}^T d\Omega_0 = \int_{\Omega_0} \partial \mathbf{h} \mathbf{S} \mathbf{F}^T d\Omega_0 \quad (4)$$

The above integral needs to be derived numerically.

Consider a continuous function ϕ of natural element coordinates ξ, η, ζ :

$$\int_{\Omega} \phi(\xi, \eta, \zeta) d\Omega = \int \int \int_{-1}^1 \phi(\xi, \eta, \zeta) d\xi d\eta d\zeta \quad (5)$$

This integral can be approximated via three-dimensional Gaussian quadrature integration with quadrature points in

each dimension resulting in a triple summation of three-dimensional weights multiplied with the function value at each quadrature point. This triple summation is often simplified to a single summation where the product of weights in each dimension are multiplied yielding one single weight for n_q quadrature points:

$$\int_{\Omega} \phi(\xi, \eta, \zeta) d\Omega \approx \sum_{q=1}^{n_q} w_q \phi(\xi_q, \eta_q, \zeta_q) \quad (6)$$

For tetrahedral elements, the Jacobian:

$$J = \det\left(\frac{\partial X}{\partial e_c}\right) = 6V_0 \quad (7)$$

for element natural coordinates $e_c = \{\xi, \eta, \zeta\}$ and the volume of the undeformed tetrahedron V_0 . This yields the following expression for:

$$d\Omega_0 = \frac{1}{6} J d\xi d\eta d\zeta = V_0 d\xi d\eta d\zeta \quad (8)$$

Substituting Eq. 8 into Eq. 4 and using Eq. 6 to perform numerical integration, we get:

$$f = \int \int \int_0^1 V_0 \partial \mathbf{h} \mathbf{S}^T d\xi d\eta d\zeta \approx \sum_{q=1}^{n_q} w_q V_0 \partial \mathbf{h} \mathbf{S}^T \quad (9)$$

Since the above integration is in the range $[0, 1]$ instead of the canonical $[-1, 1]$, the weights in each dimension are scaled to 1. The combined weight w_q across the three dimensions will then also be equal to 1 thus Eq. 9 reduces to:

$$f = V_0 \partial \mathbf{h} \mathbf{S}^T \quad (10)$$

The deformation gradient \mathbf{F} in Eq. 10 can be derived from:

$$\mathbf{F} = \partial \mathbf{h} \mathbf{u} + \mathbf{I} \quad (11)$$

with the identity matrix \mathbf{I} .

Finally, to derive $\partial \mathbf{h}$, we apply the chain rule to the matrix of shape function derivatives:

$$\partial \mathbf{h} = \frac{\partial N}{\partial X} = \frac{\partial N}{\partial e_c} \frac{\partial e_c}{\partial X} = \frac{\partial N}{\partial e_c} \left(\frac{\partial X}{\partial e_c} \right)^{-1} \quad (12)$$

The shape functions for a constant strain tetrahedral element are:

$$N_1(\xi, \eta, \zeta) = \xi \quad (13)$$

$$N_2(\xi, \eta, \zeta) = \eta \quad (14)$$

$$N_3(\xi, \eta, \zeta) = \zeta \quad (15)$$

$$N_4(\xi, \eta, \zeta) = 1 - \xi - \eta - \zeta. \quad (16)$$

The Jacobian matrix \mathbf{J} is given by:

$$\mathbf{J} = \frac{\partial X}{\partial e_c} = \mathbf{X} \frac{\partial N}{\partial e_c} = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -1 & -1 \end{bmatrix} \quad (17)$$

where \mathbf{X} is the matrix of nodal coordinates. Working out Eq. 17 gives:

$$\mathbf{J} = \begin{bmatrix} x_1 - x_4 & x_2 - x_4 & x_3 - x_4 \\ y_1 - y_4 & y_2 - y_4 & y_3 - y_4 \\ z_1 - z_4 & z_2 - z_4 & z_3 - z_4 \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ E_3 \end{bmatrix}^T \quad (18)$$

where E_i are column vectors representing tetrahedral edges.

We can now derive $\partial \mathbf{h}$ by substituting Eq. 18 (following inversion) into Eq. 12:

$$\partial \mathbf{h} = \frac{\partial N}{\partial e_c} [E_1 \ E_2 \ E_3]^{-1} \quad (19)$$

We will relate the internal forces described in Eq. 10 with the contact forces derived later in Sect. 3.3.

3.2 Mechanical contact method

The forces which affect the motion of the baby's head (CMs) are a result of the contact interaction between the baby head and the maternal pelvic anatomy—including the bony pelvis, the pelvic floor muscles and ligaments and the fully dilated uterine cervix—during the second stage of labour which is the expulsion stage. During the first stage of labour, the movement of the baby is slow and consists of the head being in contact with the uterine cervix (i.e. the lower part of the uterus) which dilates from almost closed to full dilation (approx. 10 cm on average). The fetal head to uterine cervix contacts causes the phenomenon known as fetal head moulding (Lapeer and Prager 2001). We ignore fetal head moulding at this stage of the development thus consider the fetal head to be rigid. Thus, the simulation starts at the end of the first stage or start of the second stage. The fetus is descending and is not yet in contact with the pelvic anatomy. To detect contact, we need to perform a *contact detection* procedure which is better known in computer graphics and computer games as *collision detection*. Once contact/collision is detected between the baby head and any part of the maternal anatomy, contact pairs need to be established between the two surfaces according to a *contact discretisation* method. Once contact pairs have been established, the contact needs to be 'resolved' using a *contact resolution* method. We will discuss each of these three methods next. One should keep in mind that these procedures are part of a continuous update loop.

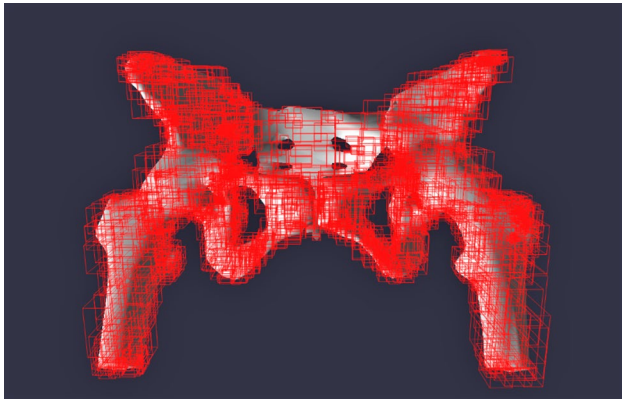


Fig. 1 The bony pelvis model with octree-based AABBs (axis aligned bounding boxes)

3.2.1 Contact detection

We used hierarchical collision detection (CD) to minimise processing overhead using a two-step process with a broad and narrow phase.

During the *broad phase*, FE mesh models are first subdivided in different rectangular regions using a Bounding Volume Hierarchy (BVH) which in our implementation is an octree subdivision. Figure 1 shows an octree built around the bony pelvis model. BVHs can be implemented using pointers, but since elements are not typically removed in our application, this approach causes a significant overhead when the tree needs to be traversed during the CD process due to indirect memory access. Therefore, we adopted a pointer-less method where leaf nodes are stored in sequential order and their position being accessed through an indexing function. We used Morton code or z-order curve indexing for this purpose (Morton 1966) which reduces the dimensionality of the key from three to one. The construction of the octree was done in a top-down fashion, i.e. starting from the entire mesh model and subdividing in smaller boxes until the smallest box contains a preset maximum number of faces. This approach is preferred over a bottom-up approach which starts at the level of a single face. Although both approaches are $O(N)$ for N faces, the latter requires to run a nearest neighbour test at each new level up the tree which causes a significant overhead for complex FE meshes. The broad phase at runtime uses a tree traversal algorithm. Since some of the meshes are deformable, e.g. the pelvic floor muscle mesh (see Fig. 12), the orientation of the AABBs (axis aligned bounding boxes) is relatively changed to one another which turns them in OBBs (Oriented Bounding Boxes). Even a rigid body in the simulation, e.g. the fetal head, will change in global directions so if AABBs were referred to a global coordinate system they will turn into OBBs as well. There are two ways to resolve this: either both OBBs to be

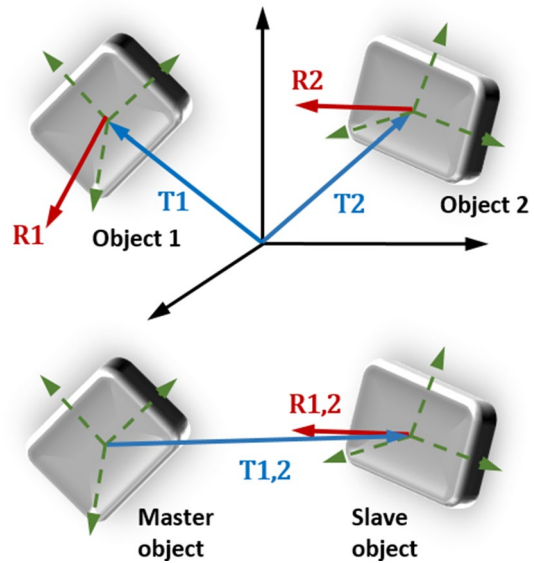


Fig. 2 Top: OBBs of object 1 and 2 referred to global coordinate system. Bottom: Object 1 becomes master object, and object 2's position and orientation are referred to the master object's local coordinate system

checked for collision are referred to a global coordinate system or the local coordinate system of one of the two bodies (typically the rigid body) is used as the master coordinate system and the other body's OBBs are referred to it. Figure 2 shows the two approaches and illustrates that the second approach is favourable as only one transformation is required which results in a speedup for high OBB counts, typical for complex FE mesh traversal.

Once potential collision between parts of potentially colliding bodies is detected through the nearest OBBs, the *narrow phase* identifies face-to-face collision between the relatively small number of faces contained in each of the OBBs. We use the well-established SAT (Separating Axis Theorem) for this purpose (Miller 1997).

3.2.2 Contact discretisation

Ignoring multiple body contact and self-contact, we assume two bodies to be in contact at some instance in time at one (or multiple) section(s) of their respective boundaries. Since the bodies or 'objects' in our application are discretised FE mesh models, we select one object's contact surface to be the 'master' surface and the other to be the 'slave' surface. Typically, the latter is the surface with the higher mesh resolution. Due to the discrete nature of contact interaction, contact pairs between master and slave surfaces are selected according to three different scenarios. In a node-to-node (NTN) approach, these contact pairs consist of corresponding nodes on each of the two surfaces on a one-to-one basis, based on minimal distance. This works fine for

contact in the normal direction, but in the tangential direction, i.e. when slip occurs, node correspondence may be lost for some nodes. In a node-to-segment (NTS) discretisation, a node from the slave surface is paired with a segment of the master surface. This approach is better suited for large deformations and large tangential sliding as compared to the NTN approach. The method works fine if slave surfaces have higher resolution than master surfaces but if this is not the case then undetected or ‘spurious’ penetrations may occur. Finally, the segment to segment (STS) relates segments of each one surface to the other. Though attractive in theory, this method is complex to implement in practical problems. In our application, we opted for the NTS discretisation due to the large displacements between the slave surface (the fetal head) and the master surfaces (maternal anatomy including bony pelvis, pelvic floor muscles and uterine cervix).

3.2.3 Contact resolution

The body parts in contact in our application are the rigid fetal head with either the deformable pelvic floor muscles or the rigid bony pelvis. Both rigid body parts are of arbitrary shape. We do not consider friction at this stage due to the complexity of finding realistic values of friction coefficients for our particular application. As such, the following Hertz-Signorini-Moreau (HSM) contact conditions hold for frictionless contact (Yastrebov 2013) for an arbitrary rigid body with an arbitrary deformable body on the contact zone Γ_c (of which the active contact zone $\bar{\Gamma}_c$ is the subset of contact pairs in contact):

$$g \geq 0, \sigma_n \leq 0, g\sigma_n = 0, \sigma_t = \mathbf{0} \quad (20)$$

where g is the gap between corresponding elements of a contact pair. The condition that the gap should be non-negative is crucial to the underlying principles of various numerical contact methods discussed later. The contact pressure σ_n is non-positive for non-adhesive contact. The third condition signifies that a positive gap implies a zero contact pressure and the presence of a positive contact pressure, a zero gap. Finally, the tangential stress vector σ_t is the zero vector due to frictionless contact.

The SAT CD method looks for intersections at the level of the primitives which are typically outer faces of the object boundaries. Due to the *a posteriori* nature of SAT CD when using finite time steps, collisions can be missed hence causing interpenetration of the two objects. In applications such as game physics, this may be avoided by either using *a priori* CD methods which preempt collisions and avoid interpenetration or by rectifying the interpenetration by calculating the exact point of collision. The concept of interpenetration (or negative gap g) and its rectification to satisfy the HSM condition of a non-negative gap g (Eq. 20) lays at the basis of several numerical contact methods. Next, we give a

brief review of some popular contact methods followed by more in-depth coverage of the projection-based partial Dirichlet–Neumann (pDN) contact method which we adapted to our specific problem statement.

Penalty method (PM) treats interpenetrations as strict constraint violations by introducing a resistive ‘penalty’ force as a function of the gap (or penetration) between the node and the master surface. An abstract way of visualising penalty-based methods is to imagine a resistive spring between the contact surfaces which has length 0 when there is no contact and which elongates with increasing penetration (violation of the HSM first condition). The relation between the force and the gap is often taken to be linear (Hooke’s law) due to its simplicity although nonlinear (quadratic and exponential) functions provide more accurate results. To fulfil a non-penetration condition, the penalty force has to be infinite which implies that in practical solutions this condition can only be approximately fulfilled. Since the penalty parameter(s), which relate(s) the force to the gap, is/are problem specific hence empirically determined (Kikuchi and Oden 1988), they cannot be used in a different context, e.g. when scaling the problem geometry or modifying material properties or the time step.

Lagrange multiplier method (LMM) The LMM aims to fulfil the HSM conditions using constrained minimisation. The Lagrangian is given by:

$$\mathcal{L}(\mathbf{u}, \lambda_n) = \Pi(\mathbf{u}) + \int_{\bar{\Gamma}_c^1} \lambda_n g(\mathbf{u}) d\bar{\Gamma}_c^1 \quad (21)$$

where \mathbf{u} is the displacement vector on the active contact zone $\bar{\Gamma}_c^1$ of deformable object 1; $\Pi(\mathbf{u})$ is the corresponding deformation energy; λ_n are a continuous set of Lagrange multipliers on the active contact zone to enforce the gap condition $g(\mathbf{u}) \geq 0$. The stationary condition is obtained by taking the variation of $\mathcal{L}(\mathbf{u}, \lambda_n)$. It should be noted that the following relation holds for the contact pressure σ_n :

$$|\sigma_n| = |\lambda_n| = \frac{\partial \mathcal{L}}{\partial g} \quad (22)$$

Since the LMM introduces additional degrees of freedom into the solution that represent the contact forces between the bodies in contact, the solution becomes harder to obtain and may in fact become impractical to compute for relatively large problems (Pietrzak and Curnier 1999). The augmented Lagrange method (ALM) combines the LMM with the PM which leads to improved solution convergence, thus considerably improving the speed of the solver (Simo and Laursen 1992). The LMM was used with an *explicit FE* model by Taylor (1989) in the PRONTO 3D software for transient solid dynamics. Heinstein (1997) also used the LMM in

combination with a matrix-free explicit FE model using an iterative approach to enforce the contact non-penetration conditions outlined in (20). Further work by Heinsteint et al. (2000) focused on contact-impact modelling for large deformation problems (including friction) in an explicit dynamic FE setting. Despite the focus being on impact modelling, the method can be used for quasi-static contact problems as well and can deal with multiple object contact. Johnsen et al. (2012) combined the method by Heinsteint et al. (2000) with Miller’s TLED method (Miller et al. 2007) and integrated these into the NiftySim TLED simulation software (Johnsen et al. 2014). Additional functionality includes contact normal smoothing for improved stability and a friction model. Johnsen’s updated method is also capable of simulating contact between two soft tissues or soft tissues and rigid objects. Further development by Johnsen et al. (2015) includes the earlier mentioned BVH (Sect. 3.2.1) to improve performance for high polygon count meshes using a heuristic approach.

Projection-based method (PBM) The main issue with the PM is its inability to satisfy the conditions outlined in (20). Whilst the LMM-based methods do satisfy these conditions, when used within an explicit FE model, the solution depends on the time step used for explicit time integration. To deal with these shortcomings, projection-based methods treat contact conditions strictly kinematically by resolving contact by moving or ‘projecting’ violating (or interpenetrating) nodes out of penetration. As such the contact conditions (20) are satisfied, whilst the computational cost of projection is significantly smaller than for LMM-based approaches. Since PBMs are strictly kinematic, the contact force is unknown. Cirak and West (2005) use a momentum-based approach to derive the contact force, based on the momentum exchange between interacting nodes. Interpolation is used when nodes hit faces instead of other nodes. Momentum is calculated from the mass and velocity of the nodes in contact. The method is useful for impact modelling but less so for quasi-static approaches as there is little or no momentum exchange. The *partial Dirichlet–Neumann (pDN)* contact method described by Yastrebov (2013) is also projection based. The non-penetration conditions (20) are enforced by projection of the (slave) nodes onto the (master) surface using the Dirichlet boundary conditions. Tangential contributions including friction are treated as Neumann boundary conditions. The method was not originally designed for explicit FE though the use of a Dirichlet boundary condition along the normal direction of the contact surface leads to a more robust contact resolution due to having no dependency on the reaction force and the time step in explicit FE. Due to the process of childbirth being a slow quasi-static process, the relative velocities of the contact surfaces remain small and no impact is present nor do we consider any friction at this stage. As such, the adaptation of the pDN method

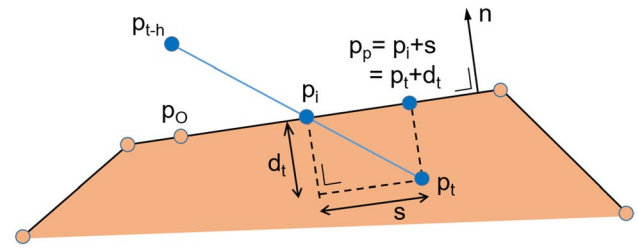


Fig. 3 The principle behind the implemented pDN method. Blue nodes are part of the slave surface penetrating the pink master surface. The node \mathbf{p} at time $t - h$ (\mathbf{p}_{t-h}) will interpenetrate the master surface at the next time step ($h = \delta t$) ending up at position \mathbf{p}_t . To satisfy the HSM conditions, \mathbf{p}_t is projected back to the surface to end up at the position $\mathbf{p}_p = \mathbf{p}_i + \mathbf{s} = \mathbf{p}_t + \mathbf{d}_t$, where \mathbf{p}_i is the intersection point and \mathbf{s} the tangential slip vector

for our purpose has resulted in the projection-based contact method algorithm illustrated in Fig. 3. The slave surface nodes are in blue colour, and the master surface is pink. The issue that arises is that in between one time step $\delta t = h$, a free node \mathbf{p} at position \mathbf{p}_{t-h} will have penetrated the surface ending up at position \mathbf{p}_t whilst violating the HSM conditions described earlier. We wish to move the node to the position $\mathbf{p}_p = \mathbf{p}_i + \mathbf{s}$ where \mathbf{p}_i is the intersection point of node \mathbf{p} with the master surface and \mathbf{s} is the tangential slip that would have occurred within the time step. Rather than calculating the exact position of \mathbf{p}_i , it is more straightforward to project \mathbf{p}_t in the direction of the master surface’s normal \mathbf{n} over the distance d_t (gap) which is obtained from:

$$d_t = (\mathbf{p}_t - \mathbf{p}_o) \cdot \mathbf{n} \quad (23)$$

where \mathbf{p}_o is an arbitrary node on the master’s intersection plane (plane origin). The projected position \mathbf{p}_p is then given by:

$$\mathbf{p}_p = \mathbf{p}_t + d_t \mathbf{n} \quad (24)$$

Algorithm 4 in “Appendix 3” further describes the implementation of the projection-based contact method.

3.3 Calculating the contact force

Since the proposed pDN method only satisfies the non-penetration conditions, the contact force needs to be calculated separately as part of the explicit FE formulation we outlined in Sect. 3.1. The FEM integrates stress over element volume to evaluate nodal forces. Here, nodal forces are evaluated across the element surface. Therefore, we adopt the finite volume method (FVM) (Teran et al. 2003) to evaluate the stress-based contact force. “Appendix 1” illustrates the basic principle of calculating a nodal force for a 2D triangular element. Extending to a 3D tetrahedron (see Fig. 4), we get for the nodal force at node i :

$$f_i = - \sum_{j=1}^n \frac{1}{3} \sigma_j (a_{j,1} \mathbf{n}_{j,1} + a_{j,2} \mathbf{n}_{j,2} + a_{j,3} \mathbf{n}_{j,3}) \quad (25)$$

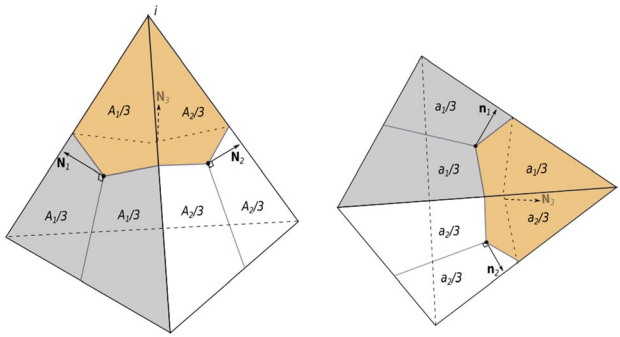


Fig. 4 FVM integration for a 3D region showing a node i and the elements of its surrounding polygon in the reference (left) and current (right) configurations. The coloured region shows the neighbourhood in which the stress is integrated. $A_j(a_j)$ and $\mathbf{N}_j(\mathbf{n}_j)$ are the area and normal, respectively, for the reference (current) configuration of face j

σ_j is the Cauchy stress in element j ; $a_{j,k}$, $\mathbf{n}_{j,k}$, $k = 1 \dots 3$ —areas and normals, respectively, of the faces of element j comprising node i (current or spatial configuration); n is the number of elements of the surrounding volume around node i . Replacing the Cauchy stress with the nominal stress and using Nanson's formula to express areas and normals in the reference (material) configuration (Holzapfel 2000) gives:

$$f_i = - \sum_{j=1}^n \frac{1}{3} \mathbf{P}_j (A_{j,1} \mathbf{N}_{j,1} + A_{j,2} \mathbf{N}_{j,2} + A_{j,3} \mathbf{N}_{j,3}) \quad (26)$$

where \mathbf{P}_j is the nominal stress tensor for element j ; $\mathbf{N}_{j,l}$, $A_{j,l}$ are the normal and area, respectively, of the node i adjacent to face l of element j in the reference configuration. The sum of the product of normals and areas can be pre-computed per element j : $b_j = \frac{1}{3} \sum_{l \neq j} A_l \mathbf{N}_l$. Additionally, for a tetrahedron, the following relation holds: $\sum_{k=1}^4 A_k \mathbf{N}_k = 0$. Putting this together, we arrive at the matrix \mathbf{B}_m :

$$\mathbf{B}_m = -\frac{1}{3} [A_1 \mathbf{N}_1 \ A_2 \mathbf{N}_2 \ A_3 \mathbf{N}_3 \ A_4 \mathbf{N}_4] = [b_1 \ b_2 \ b_3 \ b_4] \quad (27)$$

As such, force contributions of one element e to each of its nodes as derived from the FVM can be written in the nodal force contribution matrix:

$$f_{\text{FVM}}^e = \mathbf{P} \mathbf{B}_m = [g_1 \ g_2 \ g_3 \ g_4]^T \quad (28)$$

With g_k , the nodal force contributions of element e to node k . Equation 10 in Sect. 3.1 (TLED) also represents the individual element contribution to the nodal forces and has to equate in absolute value (though opposite sign) to the force derived in Eq. 28 which gives³:

$$\mathbf{P} \mathbf{B}_m = -V_0 \mathbf{F} \mathbf{S}^T \partial \mathbf{h}^T \quad (29)$$

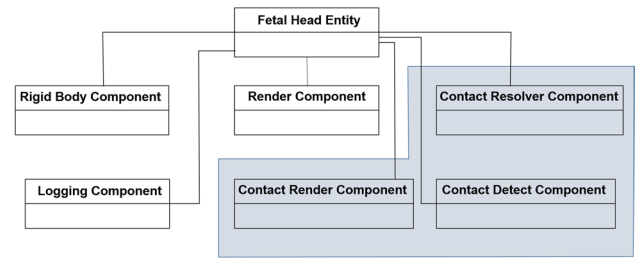


Fig. 5 An entity-component system (ECS) diagram for the fetal head

Since $\mathbf{P} = \mathbf{F} \mathbf{S}$ and $\mathbf{S} = \mathbf{S}^T$ (Holzapfel 2000), the term $\mathbf{F} \mathbf{S}^T$ in Eq. 29 can be replaced by \mathbf{P} which cancels out with its left-hand side equivalent resulting in:

$$\mathbf{B}_m = -V_0 \partial \mathbf{h}^T \quad (30)$$

3.4 Implementation

The underlying architecture of the birth simulation software is an entity-component system (ECS) (Nystrom 2014). An ECS works well in computer game styled software where many ‘entities’ use similar generic components. An *entity* identifies an object, e.g. fetal head, pelvis, pelvic floor muscle. It will typically encapsulate this object’s position, orientation, velocity and so forth, but it will not contain any methods with respect to the behaviour of the object. This is held in different (generic) *components* that relate to *systems*, e.g. systems for rendering, physics, object manipulation (UI), camera, keyboard and windows. As such, behaviour is decoupled from the actual object. Figure 5 shows an ECS diagram for the fetal head in the simulation.

The requirement for interactive (realtime) rates in our childbirth simulation dictates that the computations are to be performed within small time periods (preferably smaller than 16 ms). To facilitate this, various underlying systems need to run in parallel. Figure 6 shows a flow chart of all main systems as part of the simulation. The TLED implementation is run on the GPU using the OpenCL⁴ API. Note that other research teams have developed GPU-based implementations of the TLED method using the NVidia CUDA⁵ technology (Taylor et al. 2008; Johnsen et al. 2014). Figure 7 gives an overview of each stage of the TLED implementation. Each of the stages of pre-computation, element and node processing are described in Algorithms 1–3 in ‘Appendix 3’. The pre-computation step is executed once, whereas the nodal and elemental updates are constantly updated in

³ The proof of this equality is provided in ‘Appendix 2’.

⁴ <https://www.khronos.org/opencl/>.

⁵ <https://developer.nvidia.com/cuda-zone>.

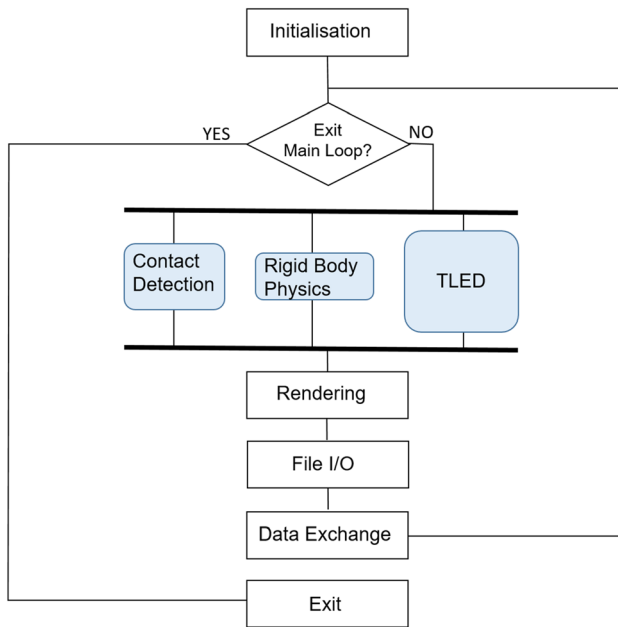


Fig. 6 Flow chart of the processes involved in the childbirth simulation. A sync event at the end of each frame enables synchronisation of the concurrent processes related to contact detection, rigid body dynamics and TLED

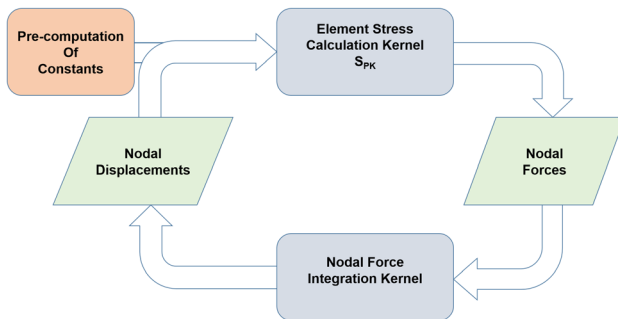


Fig. 7 The core steps of the TLED algorithm as implemented in the childbirth simulation software. The two kernels constantly exchange data. S_{PK} is the second Piola–Kirchhoff stress. See also Algorithms 1–3 in “Appendix 3”

a continuous cycle of data exchange between the two kernels. The nodal displacement updates at the next time step $t + 1$ are calculated using an explicit FE solver with Verlet numerical integration:

$$u_{t+1} = A(\mathbf{R}_t - \mathbf{F}_t) + Bu_t + Cu_{t-1} \quad (31)$$

where u is the nodal displacement, \mathbf{R}_t , the external force vector, \mathbf{F}_t , the internal force vector, both at time t and the pre-computed constants A, B, C are:

$$\begin{aligned} A &= \left(\frac{1}{\Delta t^2} \mathbf{M} + \frac{1}{2\Delta t} \mathbf{C} \right)^{-1} \\ B &= \frac{2}{\Delta t^2} \mathbf{M} \mathbf{A} \\ C &= \frac{1}{2\Delta t} \mathbf{C} \mathbf{A} + \frac{B}{2} \end{aligned} \quad (32)$$

where \mathbf{M} is the mass matrix and \mathbf{C} the damping matrix. Convergence to the correct solution is conditional to a sufficiently small time step, Δt , according to the Courant–Friedrichs–Lewy (CFL) condition (Lewy et al. 1928).

Figure 8 shows the basic class diagram of the TLED implementation.

The implementation of the pDN contact method is illustrated in Algorithm 4 in “Appendix 3”. The main nodal boundary condition (BC) used in the pDN method is a plane constraint.⁶ A penetrating slave node, with a plane constraint attached to it, will be projected onto the master surface plane which is described by the four parameters of the plane equation (step 2 in Algorithm 4). The contact detection process is executed on the CPU.

Figure 6 illustrates the parallel execution of TLED and contact detection at the abstract level.

Figure 9 shows how the contact and TLED processes interact with one another at the process level. During the *sync* event, data are exchanged between CPU (contact) and GPU (TLED). This allows TLED to use the latest contact conditions as BCs until the next *sync* event, whilst the latest nodal deformations are considered to be constant in the contact detection until the next *sync* event. The system employs *amortised* contact detection where contact is performed once per frame as opposed to performing it at every TLED subtask (red and green blocks in Fig. 9) which require a complete TLED update each. The per-frame simulation time step is 16 ms, whereas TLED subtask time steps are considerably smaller. The reason for this is to allow the system to be updated in real time as CPU-processed contact detection at every TLED subtask step is not feasible due to data transfer overheads.

4 Experiments

In this section, we cover first a number of experiments on the validation of the implemented contact method, followed by the main experiment covering the ‘acid test’ as to whether our VR childbirth simulator is capable of showing the same biomechanical behaviour of a real physiological childbirth

⁶ The software facilitates the use of other nodal BCs through displacement, linear and spring constraints.

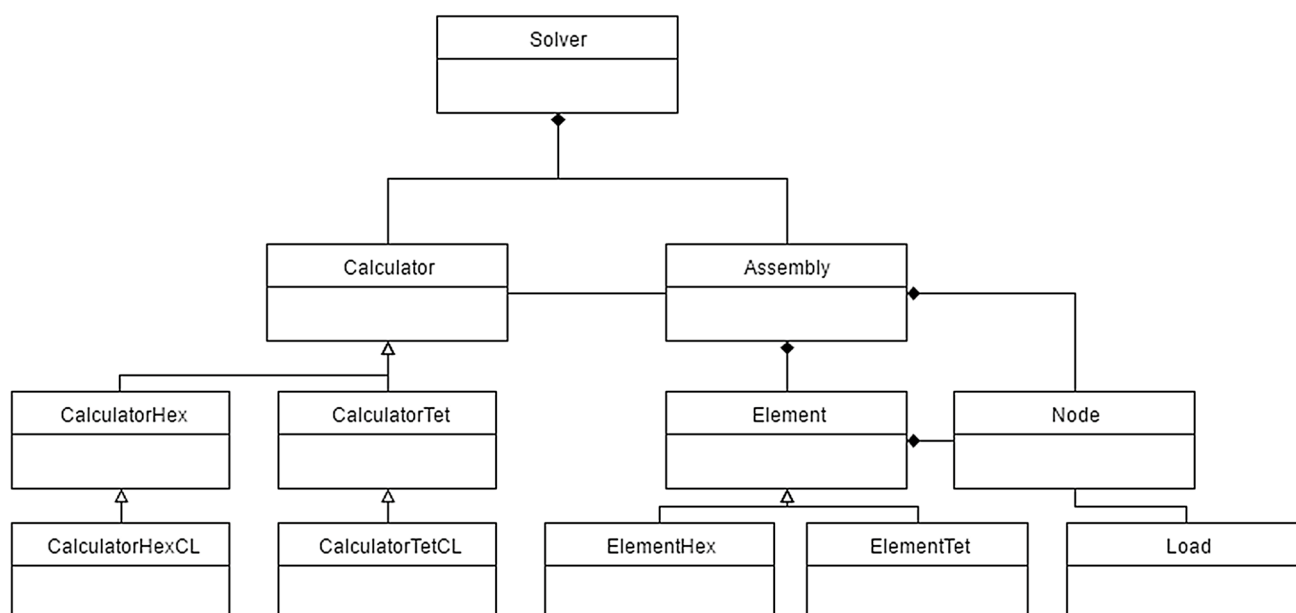


Fig. 8 Basic class diagram of the TLED implementation in the childbirth simulation software. All computations are performed in the *Calculator* subclasses, and the FE data are stored in the *Assembly* class

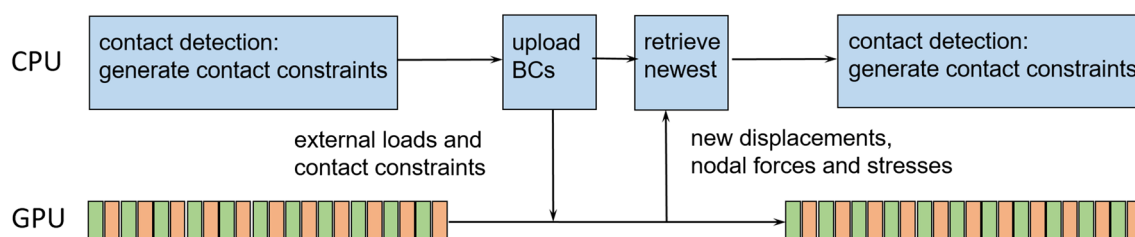


Fig. 9 The contact method executed on the CPU generates the contact boundary conditions (BCs). During the *sync* event, the data are exchanged between the CPU (contact) and GPU (TLED). Green and

red sub-tasks on the GPU as part of TLED depict per-element and per-node kernels, respectively

by exhibiting the four critical CMs of the fetal head: flexion, internal rotation, extension and external rotation.

4.1 Contact method validation

We compared the implemented projection-based contact method with TLED against the tried and tested ABAQUS software⁷ in explicit contact mode. A cube with sides of 10 cm and Neo-Hookean hyperelastic material properties was used with a bulk modulus, $k = 1$ MPa and a shear modulus, $\mu = 66$ kPa (corresponding to pelvic floor muscle tissue—see Table 3). These moduli needed to be

Table 1 Comparison of the deflection in the negative y -direction $u_{y,\max}$ in mm (note that minus signs have been omitted for clarity) for the BirthView and ABAQUS explicit FE with contact solvers, following the sphere on cube experiment

Load (kg)	#tet	BirthView	ABAQUS	Diff. (%)
10	126	17.86	14.85	− 17
	558	19.23	16.11	− 16
	1256	19.53	17.77	− 9
	5720	20.19	20.40	+ 1
	12,490	20.21	21.27	+ 5
15	126	26.74	20.04	− 25
	558	27.69	21.53	− 22
	1256	27.56	24.25	− 12
	5720	28.77	27.48	− 5
	12,490	28.72	28.42	− 1

Two spheres of 10 and 15 kg, respectively, were used. The number of elements for the cube varied between 126 and 12,490 elements

⁷ <https://www.3ds.com/products-services/simulia/products/abaqus/>.

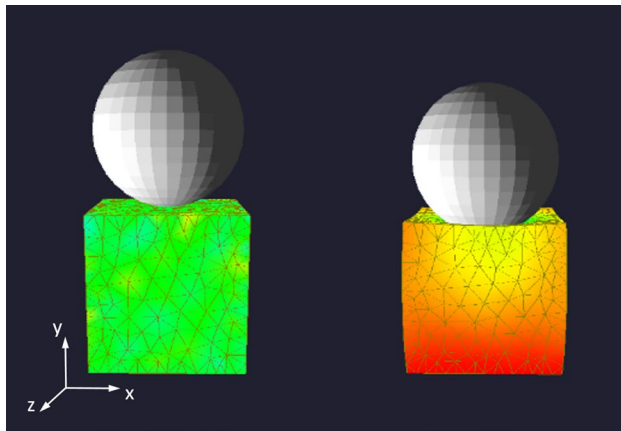


Fig. 10 A solid sphere of 10 kg is gently lowered on a hyperelastic cube (1256 elements with pelvic floor muscle tissue properties) in BirthView. All nodes on the bottom plane of the cube are encasted. Left: initial contact. Right: maximum deflection $u_{y,\max} = -19.53$ mm

Table 2 Comparison of BirthView’s penalty-based contact method and Heinsteins method on loading a hyperelastic cube with a pressure plate of 186.6 N at different values of the time step

Time step (ms)	$u_{y,\max}$ (mm)		Diff. (%)
	BirthView	Heinstein	
16	10.61	12.37	16.6
4	10.61	12.24	15.4
1	10.61	11.44	7.8
0.25	10.61	10.73	1.1
0.1	10.61	10.63	0.19
0.01	10.61	10.62	0.09

The corresponding compression values $u_{y,\max}$ in mm are shown for each of the two methods in columns 2 and 3, respectively, and the relative difference in column 4

converted into Neo-Hookean polynomial coefficients where $C_{10} = \mu/2 = 33$ kPa and $D_1 = 2/k = 0.002$ kPa⁻¹ for use in ABAQUS Explicit.

The cube consists of first-order tetrahedral elements and all bottom plane nodes are encasted. The number of elements was increased from 126 elements up to 12490 elements in five steps. A solid sphere (or ball) with a diameter of 10 cm was then gently released (initial velocity is 0) on top of the cube. Two mass densities were applied to yield spheres of 10 and 15 kg, respectively.⁸ The time step was set to 16 ms in BirthView (the name of our simulation software), whereas ABAQUS ran at a stable time step increment of 0.001 ms.

⁸ The weights have been selected on the basis of the force applied during a volitional push during childbirth which, including the weight of the baby, range approx. between a mass equivalent of 10 and 15 kg.

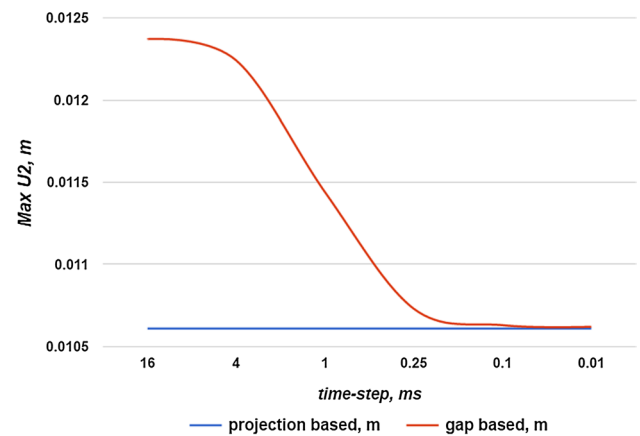


Fig. 11 Comparison of BirthView’s projection-based contact method (blue curve) and Heinsteins method (red curve) on loading a hyperelastic cube with a pressure plate of 186.6 N at different values of the time step. The abscissa shows the time step in ms, the ordinate the value of $u_{y,\max}$ in m

Table 1 shows the results.

Figure 10 shows the (10 kg) sphere with initial contact on the top surface on the cube (left) and at maximum deflection $u_{y,\max}$ in the negative y direction (right).

The results show that the BirthView physics engine (TLED/pDN) exhibits less sensitivity to the number of elements used as compared to ABAQUS explicit contact which underestimates the deflection at lower element counts. This phenomenon is clearer at the higher weight of 15 kg (which is the mass equivalent to the force of a combined volitional push and the weight of the baby) where the percentage difference between the deflection at 126 elements versus 12490 elements is just 7% in BirthView, whereas it is almost 30% in ABAQUS. Ignoring this trend, the deflection at the higher element counts corresponds well between BirthView and ABAQUS with differences of no more than 5%.

4.2 Time step sensitivity

To illustrate the relative insensitivity to the time step of the penalty-based method, implemented in BirthView, a comparison to Heinsteins method (Heinstein et al. 2000) is facilitated through the following experiment: The same cube with sides of 10 cm and pelvic floor muscle hyperelastic properties is used. A square plate of 10 × 10 cm is gently lowered on the cube and the deflection, $u_{y,\max}$, in the negative y direction is measured for different time steps. Table 2 and Fig. 11 show the results. It is clear that Heinsteins method only converges to the true solution when the time step is sufficiently small, whereas the penalty-based method exhibits the same solution across all tested time step magnitudes starting from the 16 ms upper bound.

Table 3 Geometrical components of the childbirth simulation

Component	#elements	Material property	Boundary conditions
Fetal head	7900 triangular shell	Rigid	Connected to neck
Fetal neck	Two springs	Linear 100 N/m Torsional 100 Nm/rad	Fixed to fetal head and trunk Rotations in sagittal and coronal planes are constrained by head to body contact Rotations in the transverse plane are constrained by the torsional spring
Fetal trunk	1216 triangular shell	Rigid	Free moving
Bony pelvis	14,000 triangular shell	Rigid	Encastred (immobile)
Uterine cervix	1,024 tetrahedral	Neo-Hookean HE: $m = 66$ kPa, $K = 1$ MPa	Fixed in x, y, z at pubic bone
Pelvic floor muscles	28,700 tetrahedral	Neo-Hookean HE: $m = 66$ kPa, $K = 1$ MPa	Encastred—see Fig. 12
Sacrospinous ligaments	12,320 tetrahedral	Neo-Hookean HE: $m = 160$ kPa, $K = 1$ MPa	Encastred—see Fig. 15

All elements are first-order. Material properties of the soft tissues are adopted from Hoyte et al. (2008)

4.3 Childbirth simulation

Due to the complex nature of combining TLED and contact interaction algorithms, described earlier, an incremental approach was needed to arrive at the current childbirth simulation which we present next. The various experiments that laid the foundation of the simulation were successfully completed and validated and can be found in Gerikhanov (2017). The current version is capable of simulating physiological childbirth with the baby in occiput anterior (OA) position. OA implies that the back of the baby's head presents to the front of the maternal pelvis near the time of expulsion and is the most common presentation (Williams Obstetrics 2014). At the start of the second stage of labour, the back of the baby's head either faces left from the mother's viewpoint (LOA) or right (ROA).⁹

The simulation's components and their material properties (Hoyte et al. 2008) are listed in Table 3. Figure 12 shows the pelvic floor muscle mesh and the encastre points connecting the muscle to the bony pelvis. The latter was derived from the Visible Female (US national library of medicine 1996), whereas the pelvic floor muscles were derived from a 22-year-old subject of the BodyParts3D library (Mitsuhashi et al. 2009). The fetal head is a decimated and adapted version of the detailed fetal skull model used in Lapeer and Prager (2001) and is shown in Fig. 13. It is attached to the fetal trunk via a stiff combined linear and torsional spring—see Fig. 14. The fetal trunk was obtained from MR scans of a stillborn baby. The sacrospinous ligaments are triangular in

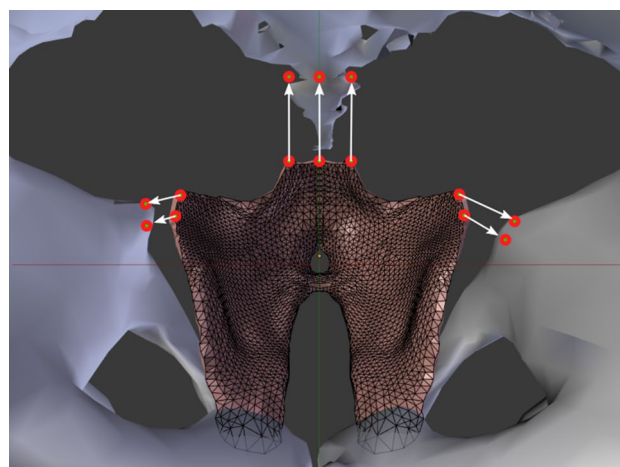


Fig. 12 The deformable pelvic floor mesh and its encastre points to connect the assembly to the maternal bony pelvis

shape and connect, on both left and right sides, through their triangular base to the edge formed by the sacrum and coccyx and their triangular apex to the ischial spine—see Fig. 15. Both the sacrospinous ligaments and the uterine cervix were modelled manually using anatomical images (Drake et al. 2014) and the Blender software (Blender Online Community 2015).

The intra-uterine pressure (IUP) lays at the basis of the uterine expulsion force which applies to the fetal buttocks and is at its peak once the uterine cervix has fully dilated which marks the end of the first stage of labour and the start of the second stage of labour. The IUP varies periodically over a period of approximately 3 min with a basal and peak pressure. Quantitative experiments on the intra-uterine pressure (IUP) have been performed since the 1950s (Turnbull

⁹ The current simulation can be run in real time on a quadcore (2.9 GHz per core) laptop or desktop with at least 16 GB (1866 MHz) of memory and a high-end video adapter card with at least 4 GB GDDR5 dedicated memory.

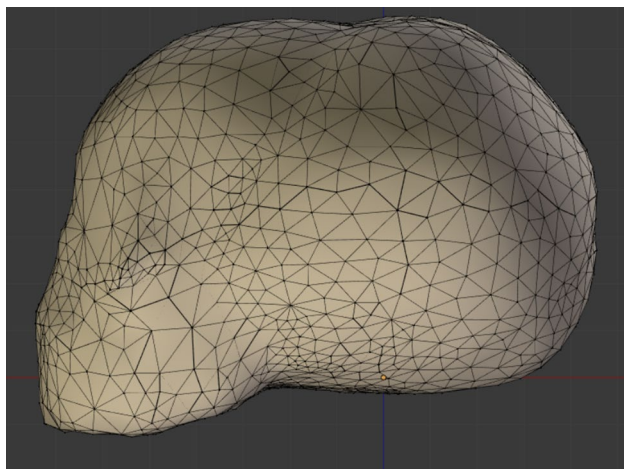


Fig. 13 Rigid fetal head model with 7900 triangular first-order shell elements

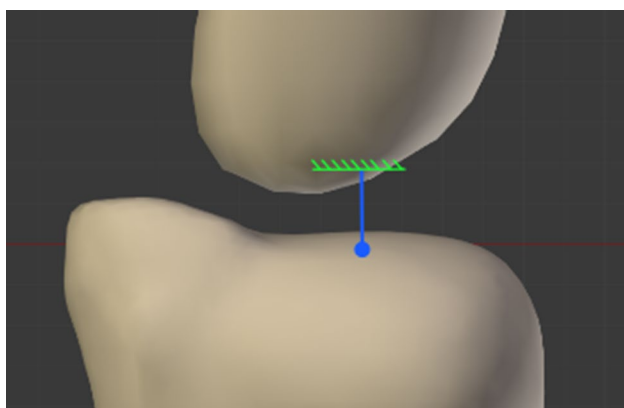


Fig. 14 A combined linear and torsional (stiff) spring connects the fetal head to the fetal trunk and functions as the fetal neck

1957). For our experiment, we adopted more recent values as reported by Ashton-Miller and DeLancey (2009), i.e. a baseline force of 16 N at rest, 54 N during a uterine contraction (peak) and 120 N during a volitional push. The initial position of the fetal head is above the pelvic brim upon which the uterine force is applied and the TLED/contact algorithm enters in an update loop, as previously shown in Figs. 6, 7 and 9, to, respectively, detect contact between head and pelvis, resolve the contact and update the deformation of the soft tissues and rigid body mechanics of the fetal head, neck and trunk. The main objective of the experiments is to observe the critical cardinal movements of the fetal head which occur in the vast majority of physiological ('natural') childbirths, i.e. in the order: flexion, internal rotation, extension and external rotation. Figure 16 shows the position, flexion (rotation in the sagittal plane) and rotation (in the transverse plane) at each stage of the expulsion of the

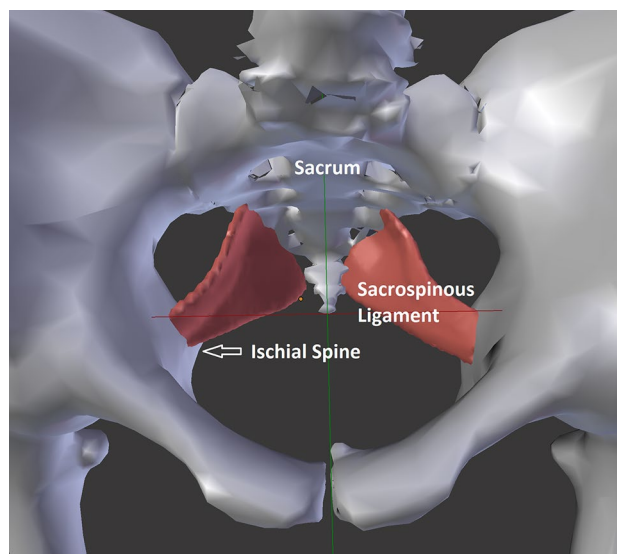


Fig. 15 The sacrospinous ligaments mesh model containing 12,320 tetrahedral first-order elements. The frontal part of each ligament is encasted to the ischial spines, whereas the back part is encasted to the sacrum

virtual baby. Flexion (negative 'deflexion'—orange curve) starts around the 36 s mark followed by internal rotation (red curve) reaching a maximum around the 120 s mark, followed by extension (or deflexion—orange curve) just above the 144 s mark followed by full external rotation around the 180 s mark. Figure 17 shows snapshots of the simulation at the four distinct stages of flexion, internal rotation, extension and external rotation.¹⁰

5 Discussion

The experiment reported in Sect. 4.3 included all soft tissues currently modelled, i.e. the pelvic floor muscles (aka Levator Ani Muscle), the sacrospinous ligaments and the uterine cervix. From preliminary experiments, not fully reported here, we started with a bony pelvis, fetal head and uterine cervix only, then added each of the soft tissues separately. Table 4 shows which of the critical CMs were observed in each of these configurations. The final configuration (Case 4) corresponds to the experiment reported in Sect. 4.3 where all soft tissues have been added. It can be seen that only in this case are all the critical CMs observed. It is interesting to note that this does not happen if either the pelvic floor muscles (Case 2) or the sacrospinous ligaments (Case 3) are present. If the latter is added (Case 3), it makes no

¹⁰ A video showing the full process is available as additional material.

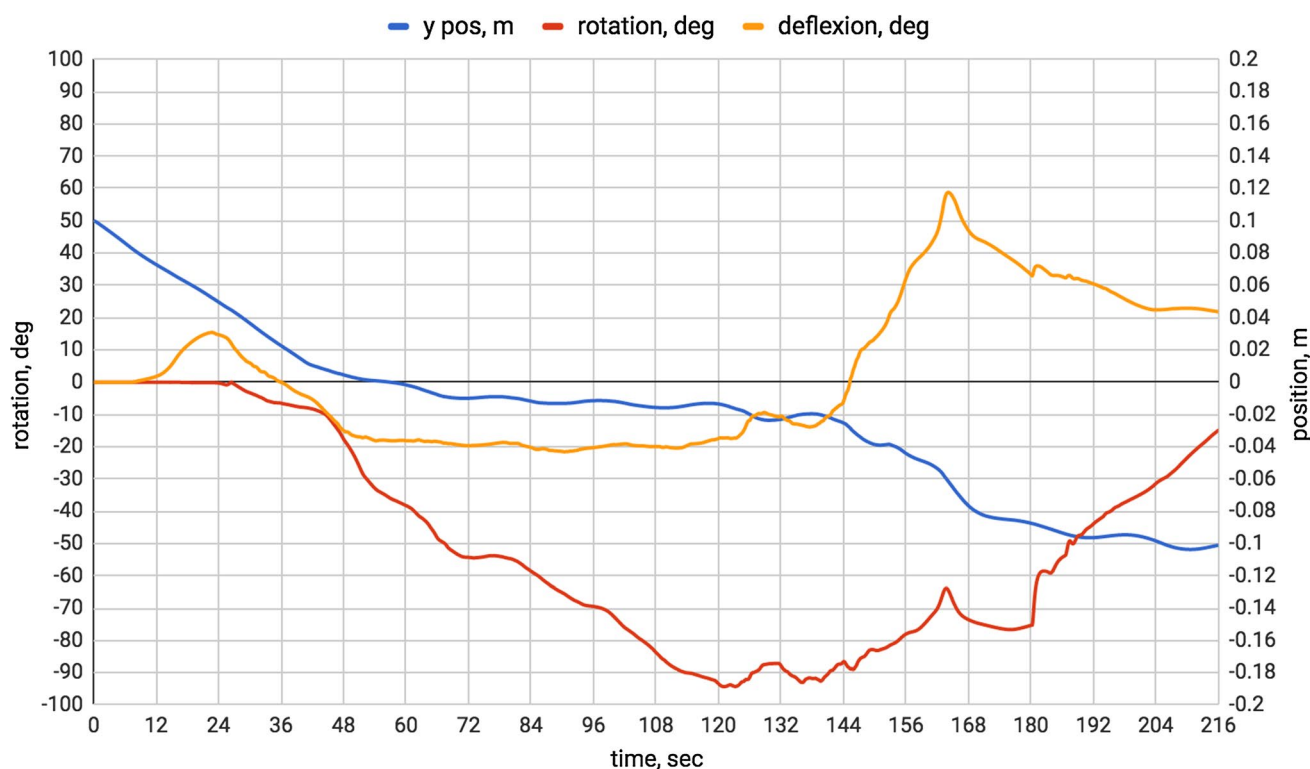


Fig. 16 The fetal head position (blue curve), rotation in the sagittal plane [orange curve: flexion (–) and extension/deflexion (+)] and rotation in the transverse plane (red curve) throughout the simulation

starting with the fetal head above the pelvic brim in ROA position and ending with the head being expelled

difference at all as the outcome is the same as in Case 1 (neither pelvic floor muscles nor sacrospinous ligaments). For Case 2 (pelvic floor muscles only), flexion is observed and internal rotation goes halfway but then does not progress any further so the virtual fetus never gets delivered. It is only when both pelvic floor muscles and sacrospinous ligaments are added that all CMs are observed and the virtual fetus is successfully delivered. This can also be observed from the trajectories shown in Fig. 16 and the screen shots from BirthView in Figs. 17 and 18. A well-known phenomenon during physiological childbirth is the fetus moving upwards and downwards (bouncing) due to the pulsating expulsive force even though the net motion is downwards (Bamberg et al. 2012). This phenomenon is also present in the BirthView Case 4 simulation and can be most clearly observed during the fetal head's extension phase as illustrated in the supplementary video and from the wavy patterns in the blue curve in Fig. 16 which corresponds to the fetal head position or 'station'. To assess robustness, Case 4 experiments with realistic variations in the initial position, the fetal head

and pelvic geometry (Hall et al. 2007), were run (but not reported here) and the four critical CMs were still observed resulting in the delivery of the virtual fetus.¹¹

In Fig. 16, we observe that the total simulation time is approx. 216 s which is just under 4 min. This is much faster than the time the second stage of labour on average lasts during a real childbirth which is approx. 20 min to 2 h (Williams Obstetrics 2014). There are several reasons for this. Firstly, various surrounding maternal organs (e.g. the bladder) have been ignored in the model as they do not actively participate though they do indirectly constrain the passage. Secondly, there will be varying degrees of friction at times due to variation in fluid content between contact surfaces. Finally, the pelvic floor muscle model in the simulation has strictly hyperelastic properties and no visco-elastic properties that will slow down progress of the simulated childbirth process.

¹¹ The uterine cervix only affects flexion at the start of the process by pre-flexing the head. It does not affect the other cardinal movements due to it being fully dilated.

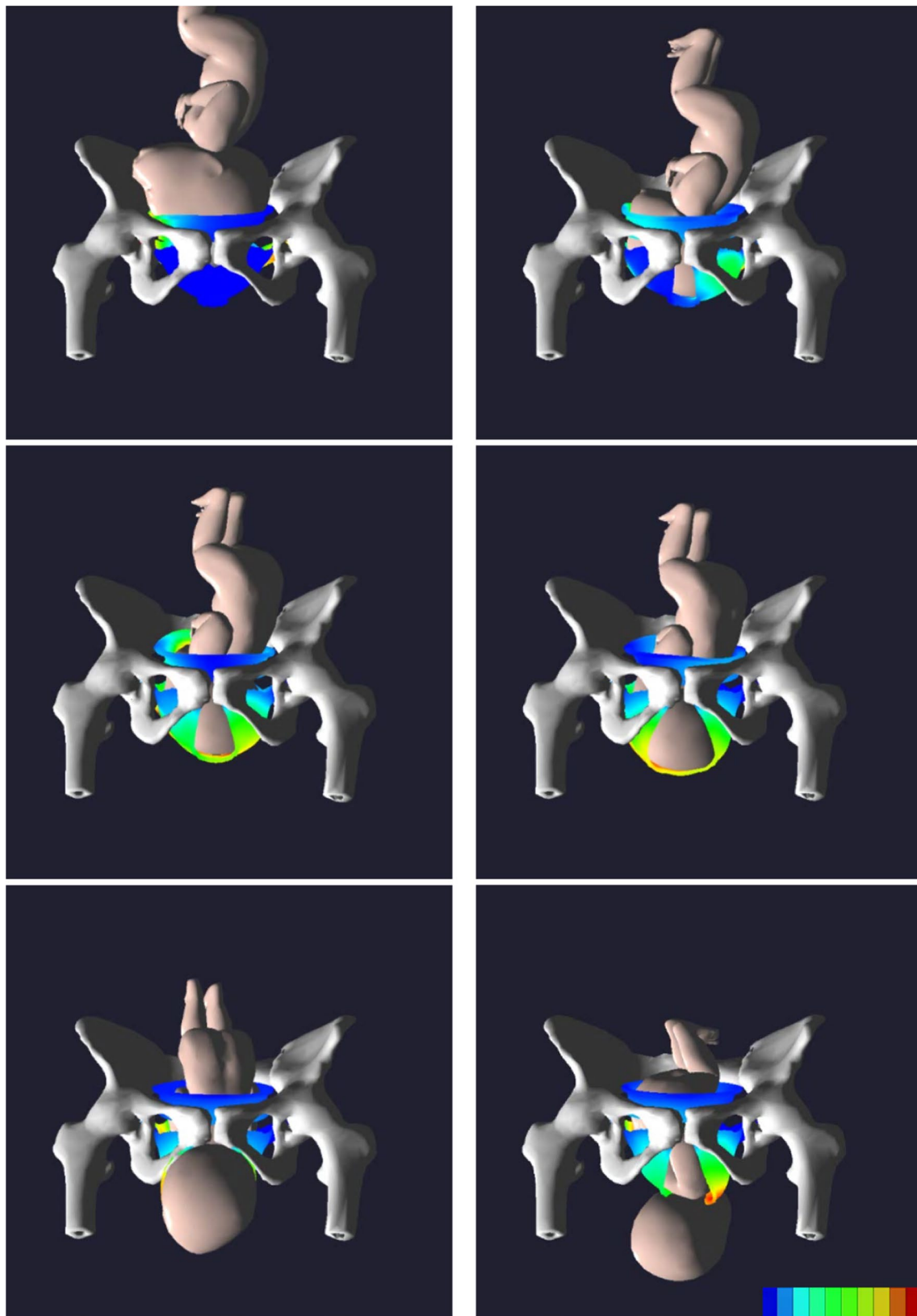


Fig. 17 A VR childbirth simulation run in BirthView: top left: initial (LOA) position; top right: flexion; middle left: internal rotation; middle right: extension start; bottom left: extension finish; bottom right:

external rotation. The colour code encodes the displacement of the soft tissues where blue starts at 0.0 mm, then 6.4, 12.8 (cyan), 19.2, 25.6 (green), 32.0, 38.4, 44.8 (yellow), 51.2 (orange) and 57.6 mm (red)

6 Conclusion

We have presented a methodology to facilitate a VR computer-based physiological childbirth simulation. We described the underlying model, including the mathematics, soft tissue models and processing using TLED, the projection-based contact method and their implementation on the GPU. It was shown from quantitative (head rotation values) and qualitative (observation of the critical cardinal movements) results that the fetal motions in the VR-based simulation are remarkably similar to its motions in a real scenario.

The current simulator which uses average-sized models of fetal and maternal anatomy will be further developed for educational purposes. From our conversations with health professionals, it has become apparent that trainee midwives and obstetricians could greatly benefit from a tool that allows them to look inside an otherwise largely non-transparent process. To arrive at such a tool, more complex scenarios should be tested. First of all, further stability testing is needed (which are already ongoing at the time of writing) to assess variations in the geometrical size and shape of key anatomical components (e.g. smaller or larger than average fetal head, different types of pelvis) and the material properties of the soft tissues (looser or stiffer cervix, pelvic floor muscles and sacrospinous ligaments). Secondly, different positions of the fetus at the start of the second stage of labour should be tested. The most important one would be shoulder dystocia (SD) as it is one of the most critical situations that can occur during childbirth in developed countries (Crofts et al. 2006). The simulator could also be used in conjunction with a force-feedback hardware device to do training of instrumental delivery such as obstetric forceps (Moreau et al. 2007).

Additional improvements, of a biomechanical nature, to potentially enhance soft tissue behaviour and interaction include the addition of visco-elastic material properties to the soft tissues to improve realism in terms of duration, a deformable fetal head including moulding (Lapeer and

Prager 2001), and articulated arms and legs to simulate the delivery of the fetal trunk once the head has been delivered. In the longer term additional maternal anatomical models should be added as well, i.e. a full uterus and abdominal organs such as the bladder.

Ultimately, the simulator could be used to predict adverse outcomes prior to the actual childbirth. This requires the currently ‘average-sized’ simulator to be transformed into a ‘patient-specific’ sized simulator. Although this is perfectly doable with modern day medical imaging technology, the following challenges would have to be dealt with:

- Key anatomical components of the mother and fetus would be segmented from high-resolution and correctly weighted MR images obtained a number of weeks before the predicted delivery. This would also require the scaling of the fetal anatomy to conform with the expected size at birth. Growth charts could be used for this purpose.
- The average models can be warped through non-rigid registration algorithms to correspond with the shape and size of the patient-specific models (including the additional scaling mentioned before). Lapeer et al. (2009) developed a fast non-rigid registration algorithm on the GPU that registers high-resolution medical image volumes in less than one second.
- To assess the material properties of the maternal soft tissue, ultrasound elastography or MRE (Magnetic Resonance Elastography) could be used.

Further applications of the simulator could include the assessment of structural changes to soft tissues at the mechanobiology level due to excessive deformation potentially followed by permanent tissue damage. In the case of the pelvic floor muscles, this could result into post-partum incontinence, whereas excessive moulding of the fetal head could cause intracranial haemorrhage(s).

Table 4 Four experiments with different configurations of soft tissues added

Case	Bony pelvis	Uterine cervix	Pelvic floor muscles	Sacrospinous ligaments	Flexion	Internal rotation	Extension	External rotation
1	Y	Y	N	N	N	~ 1/3	–	–
2	Y	Y	Y	N	Y	~ 1/2	–	–
3	Y	Y	N	Y	N	~ 1/3	–	–
4	Y	Y	Y	Y	Y	Full	Y	Y

Case 4 corresponds to all soft tissues added which is the only configuration that exhibits all four critical cardinal movements (CMs) thus leading to successful delivery of the virtual fetus. The other cases grind to a halt at the internal rotation stage which never fully completes and consequently does not allow the VR fetus to progress beyond that stage

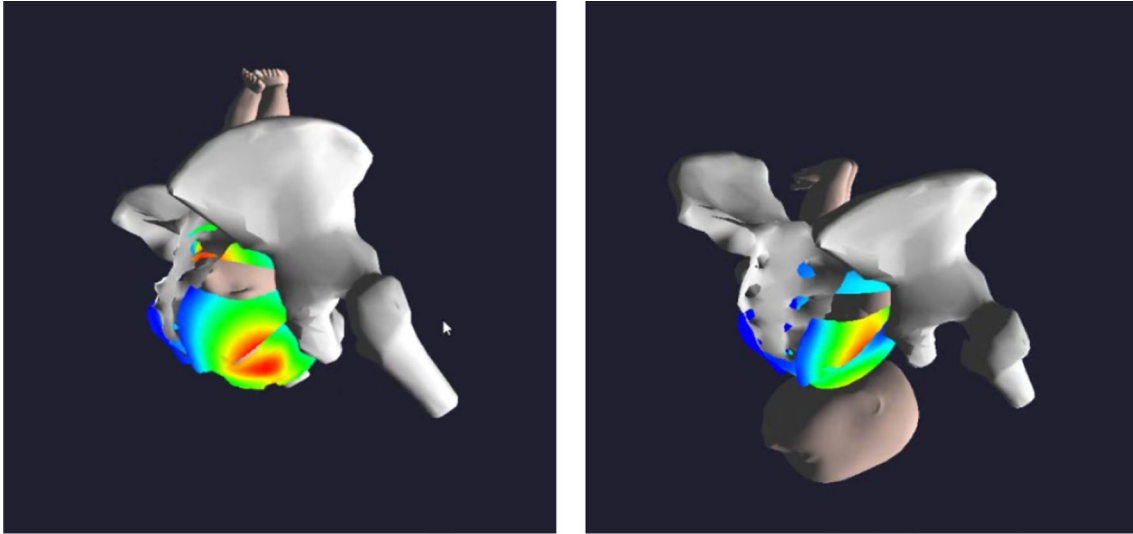


Fig. 18 Snapshots of the same simulation as shown in Fig. 17 showing the effect on the sacrospinous ligaments at the stages of internal rotation (left) and external rotation (right)

Acknowledgements This study was partially funded by the National Institute for Health Research (Grant Number HTD-515).

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix 1: Nodal contact force for 2D triangular element

Here, we describe the underlying fundamentals of the FVM (finite volume method) for later calculation of the nodal contact force for 3D tetrahedrons using a simplified case for 2D triangles first. Consider the divergence (Gauss') theorem in 2D:

$$\int_A (\nabla \cdot \sigma) dA = \oint_s (\sigma \cdot \mathbf{n}) ds \quad (33)$$

In a linear (constant strain) element, the Cauchy stress σ is constant over the element thus:

$$\nabla \cdot \sigma = 0 \quad (34)$$

This implies that the surface integral in the left-hand side of Eq. 33 is zero, and hence for the 2D triangular element shown in Fig. 19 we obtain:

$$\int_{\partial\Omega_1} \sigma \mathbf{n} ds + \int_{\partial\Omega_2} \sigma \mathbf{n} ds + \int_{\partial T_1} \sigma \mathbf{n} ds + \int_{\partial T_2} \sigma \mathbf{n} ds = 0 \quad (35)$$

Which can be rewritten as:

$$\int_{\partial\Omega_1} \sigma \mathbf{n} ds + \int_{\partial\Omega_2} \sigma \mathbf{n} ds = - \int_{\partial T_1} \sigma \mathbf{n} ds + \int_{\partial T_2} \sigma \mathbf{n} ds \quad (36)$$

The nodal force f_i is derived by applying Eq. 36 for all the surrounding elements of node i :

$$f_i = \oint_{\partial\Omega_1} \sigma \mathbf{n} ds = - \sum_j \int_{\partial T_j} \sigma \mathbf{n} ds \quad (37)$$

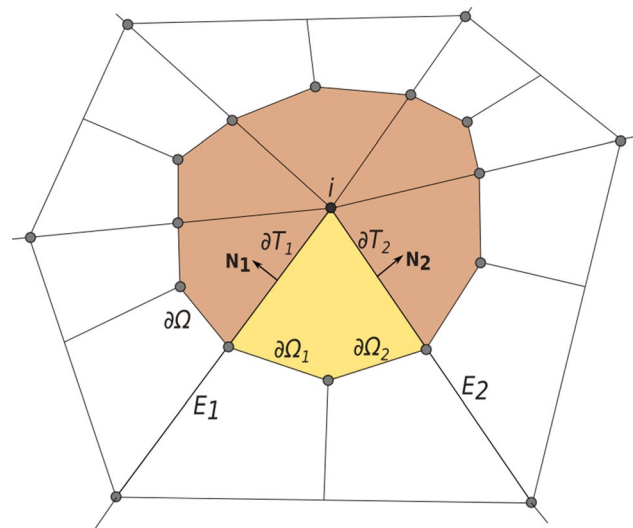


Fig. 19 FVM integration for a 2D region showing a node i and the elements of its surrounding polygon. The coloured region shows the neighbourhood in which stress is integrated. The lighter coloured triangle shows the boundary segments $\partial\Omega_j$ and edge segments ∂T_j

The length of ∂T_j is half the length of corresponding edge E_j thus for n edges with normals \mathbf{n} :

$$f_i = - \sum_{j=1}^n \frac{1}{2} \sigma(E_j \mathbf{n}_j + E_{(j+1) \bmod n} \mathbf{n}_{(j+1) \bmod n}) \quad (38)$$

Appendix 2: Proof of contact force equality and reaction forces

We assume that the forces in Eqs. 10 and 28 are equal and thus lead to Eq. 29 and subsequently Eq. 30. The proof is based on evaluating the matrix of shape function derivatives $\partial \mathbf{h}$ in function of tetrahedral properties. From the Jacobian matrix in Eq. 18, we can derive the inverse Jacobian matrix:

$$\mathbf{J}^{-1} = \frac{\partial e_c}{\partial X} = \frac{1}{6V_0} \begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{bmatrix} \quad (39)$$

It can be verified that a_i, b_i, c_i correspond to the x, y, z components, respectively, of the normal vector to the tetrahedral triangular face opposite to vertex i . This implies that the area of triangular face i is equal to:

$$A_i = \frac{1}{2} \sqrt{a_i^2 + b_i^2 + c_i^2} \quad (40)$$

and that the determinant of \mathbf{J} is six times the volume of the tetrahedron.

If we consider the vector $\mathbf{s}_i = [a_i \ b_i \ c_i]$ and the normalised vector $\hat{\mathbf{s}}_i = \mathbf{s}_i / |\mathbf{s}_i|$, where $\hat{\mathbf{s}}_i = \mathbf{N}_i$ and $|\mathbf{s}_i| = A_i$, we get:

$$\mathbf{s}_i = 2A_i \mathbf{N}_i \quad (41)$$

We can now rewrite Eq. 39 as:

$$\frac{\partial e_c}{\partial X} = \frac{1}{3V_0} \begin{bmatrix} A_1 \mathbf{N}_1 \\ A_2 \mathbf{N}_2 \\ A_3 \mathbf{N}_3 \end{bmatrix} \quad (42)$$

Substituting Eq. 42 into Eq. 12 and substituting $A_4 \mathbf{N}_4 = -(A_1 \mathbf{N}_1 + A_2 \mathbf{N}_2 + A_3 \mathbf{N}_3)$ we get:

$$\partial \mathbf{h} = \frac{1}{3V_0} \begin{bmatrix} A_1 \mathbf{N}_1 \\ A_2 \mathbf{N}_2 \\ A_3 \mathbf{N}_3 \\ A_4 \mathbf{N}_4 \end{bmatrix} \quad (43)$$

Rewriting Eq. 27 and relating it to Eq. 43, we get:

$$\mathbf{B}_m = -\frac{1}{3} \begin{bmatrix} A_1 \mathbf{N}_1 \\ A_2 \mathbf{N}_2 \\ A_3 \mathbf{N}_3 \\ A_4 \mathbf{N}_4 \end{bmatrix}^T = -V_0 \partial \mathbf{h}^T \quad (44)$$

Which is equal to Eq. 30. \square

Reaction forces

Nodal forces contributed by element e are evaluated as:

$$f_e = \mathbf{P} \mathbf{B}_m = \frac{1}{3} \mathbf{P} \begin{bmatrix} A_1 \mathbf{N}_1 \\ A_2 \mathbf{N}_2 \\ A_3 \mathbf{N}_3 \\ A_4 \mathbf{N}_4 \end{bmatrix}^T \quad (45)$$

The nominal stress \mathbf{P} relates to the traction force \mathbf{T} and area normal \mathbf{N} as $\mathbf{T} = \mathbf{P} \mathbf{N}$. The force acting on area A (reference configuration) is then:

$$f = \mathbf{T} A = \mathbf{P} \mathbf{N} A \quad (46)$$

This shows that the force acting on an element e in Eq. 45 as calculated from our contact model corresponds to the general force as a result of the traction force in an element. As such, the contact reaction force for face i from elements $j = 1 \dots 4$ can be derived as:

$$R_i = f_i = \frac{1}{3} \mathbf{P} A_i \mathbf{N}_i = -\frac{1}{3} \sum_{j \neq i} \mathbf{P} A_j \mathbf{N}_j \quad (47)$$

Appendix 3: Algorithms

Algorithm 1 TLED pre-computation of constants

- 1: Calculate element volumes according to Eq.7
- 2: Calculate node masses from element volumes and material density
- 3: Calculate the matrix of shape function derivatives \mathbf{Dh} according to Eq.12 and store in the global buffer `m_Dh`
- 4: Calculate per element nodal indices based on element connectivity and store in `nodeInds`
- 5: Calculate the Verlet integration constants A, B and C - see Eq. 32) - and store in `m_ABC`

Algorithm 2 TLED element processing

- 1: **for all** elements in assembly **do**
- 2: Collect nodal displacements into `float u[4][3]` using the per element nodal indices list from `nodeInds`
- 3: Extract the shape function derivative matrix \mathbf{Dh} from the global buffer `m_Dh`
- 4: Calculate \mathbf{F} from \mathbf{u} and \mathbf{Dh} (Eq. 11)
- 5: Calculate the Right Cauchy-Green deformation tensor \mathbf{C} from \mathbf{F} ($\mathbf{C} = \mathbf{F}^T \mathbf{F}$)
- 6: Calculate the SPK stress \mathbf{S} from \mathbf{C} , \mathbf{F} and the Lamé constants (μ and λ)
- 7: Calculate the nodal force matrix \mathbf{F}_e using the SPK stress (Eqs 28 and 29)
- 8: Spread the values from \mathbf{F}_e to the global buffer `Fx` using the nodal indices `nodeInds`
- 9: Calculate the Von-Mises stress from the components of \mathbf{S} and store the value into the `m_VMS` buffer.
- 10: **end for**

Algorithm 3 TLED nodal processing

- 1: **for all** nodes in assembly **do**
- 2: Sum the total internal nodal forces from individual element contributions
- 3: Calculate the next displacement of the node, using the precomputed values stored in $\mathbf{m_ABC}$ and store in a buffer $\mathbf{m_U_new}$ (Eq. 31)
- 4: Apply the boundary conditions (Algorithm 4) for nodal displacements and add current external force loads into the external force vector \mathbf{R}
- 5: **end for**

Algorithm 4 Projection based contact

- 1: Retrieve the previous and current node positions \mathbf{p}_{t-h} and \mathbf{p}_t and the plane equation of the master surface to derive \mathbf{n} and the arbitrary point on the plane \mathbf{p}_o
- 2: Calculate the gap d_t according to Eq. 23
- 3: Find node penetration \mathbf{p}_p according to Eq. 24

References

- Ashton-Miller JA, DeLancey JO (2009) On the biomechanics of vaginal birth and common sequelae. *Annu Rev Biomed Eng* 11(1):163
- Bamberg C, Rademacher G, Güttler F, Teichgräber U, Cremer M, Bühner C, Spies C, Hinkson L, Henrich W, Kalache KD et al (2012) Human birth observed in real-time open magnetic resonance imaging. *Am J Obstet Gynecol* 206(6):505
- Belytschko T, Liu W, Moran B, Elkhodary K (2014) *Nonlinear finite elements for continua and structures*. Wiley, New York
- Blender Online Community (2015) Blender—a 3D modelling and rendering package. Blender Foundation, Blender Institute, Amsterdam. <http://www.blender.org>
- Boissonnat JD, Geiger B (1993) 3D simulation of delivery. *Proc IEEE Vis* 93:416–419
- Buttin R, Zara F, Shariat B, Redarce T (2009) A biomechanical model of the female reproductive system and the fetus for the realization of a childbirth virtual simulator. In: Annual international conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2009), pp 5263–5266
- Cirak F, West M (2005) Decomposition contact response (DCR) for explicit finite element dynamics. *Int J Numer Methods Eng* 64(8):1078. <https://doi.org/10.1002/nme.1400>
- Crofts J, Bartlett C, Ellis D, Hunt L, Fox R, Draycott T (2006) Training for shoulder dystocia: a trial of simulation using low-fidelity and high-fidelity mannequins. *Obstet Gynecol* 108(6):1477
- Drake R, Vogl AW, Mitchell AWM (2014) *Gray's anatomy for students*. Churchill Livingstone, London
- Gerikhanov Z (2017) Simulating the cardinal movements of childbirth using finite element analysis on the graphics processing unit. Ph.D. thesis, Norwich Research Park, Norwich, UK
- Gerikhanov Z, Audinis V, Lapeer R (2013) Towards a forward engineered simulation of the cardinal movements of human childbirth. *E-Health Bioeng Conf EHB 2013*:1–4. <https://doi.org/10.1109/EHB.2013.6707394>
- Hall J, Allanson J, Gripp K, Slavotinek A (2007) *Handbook of physical measurements*. Oxford University Press, Oxford
- Heinstein M (1997) An algorithm for enforcement of contact constraints in quasistatic applications using matrix-free solution algorithms. <https://doi.org/10.2172/554827>
- Heinstein MW, Mello FJ, Attaway SW, Laursen TA (2000) Contact-impact modeling in explicit transient dynamics. *Comput Methods Appl Mech Eng* 187(3):621
- Holzappel G (2000) *Nonlinear solid mechanics: a continuum approach for engineering*. Wiley, New York
- Hoyte L, Damaser MS, Warfield SK, Chukkappalli G, Majumdar A, Choi DJ, Trivedi A, Krysl P (2008) Quantity and distribution of levator ani stretch during simulated vaginal childbirth. *Am J Obstet Gynecol*. <https://doi.org/10.1016/j.ajog.2008.04.027>
- Johnsen SF, Taylor ZA, Clarkson M, Thompson S, Hu M, Gurusamy K, Davidson B, Hawkes DJ, Ourselin S (2012) Explicit contact modeling for surgical computer guidance and simulation. *Proc SPIE Conf*. <https://doi.org/10.1117/12.911787>
- Johnsen SF, Taylor ZA, Clarkson MJ, Hipwell J, Modat M, Eiben B, Han L, Hu Y, Mertzaniadou T, Hawkes DJ, Ourselin S (2014) NiftySim: a GPU-based nonlinear finite element package for simulation of soft tissue biomechanics. *Int J Comput Assist Radiol Surg* 10(7):1077
- Johnsen SF, Taylor ZA, Han L, Hu Y, Clarkson MJ, Hawkes DJ, Ourselin S (2015) Detection and modelling of contacts in explicit finite-element simulation of soft tissue biomechanics. *Int J Comput Assist Radiol Surg* 10(11):1873
- Kikuchi N, Oden JT (1988) *Contact problems in elasticity: a study of variational inequalities and finite element methods*. SIAM, Philadelphia
- Lapeer RJ, Prager RW (2001) Fetal head moulding: finite element analysis of a fetal skull subjected to uterine pressures during the first stage of labour. *J Biomech* 34(9):1125
- Lapeer RJ, Shah SK, Rowland RS (2009) An optimised radial basis function algorithm for fast non-rigid registration of medical images. *Comput Biol Med* 40(1):1
- Lewy H, Friedrichs K, Courant R (1928) Über die partiellen Differenzengleichungen der mathematischen Physik. *Mathematische Annalen* 100:32
- Libby J, Marghoub A, Johnson D, Khonsari RH, Fagan MJ, Moazen M (2017) Modelling human skull growth: a validated computational model. *J R Soc Int*. <https://doi.org/10.1098/rsif.2017.0202>
- Lien KC, Mooney B, DeLancey JOL, Ashton-Miller JA (2004) Levator ani muscle stretch induced by simulated vaginal birth. *Obstet Gynecol* 103(1):31. <https://doi.org/10.1097/01.AOG.0000109207.22354.65>
- Liu Y, Scudder M, Gimovsky M (1996) CAD modeling of the birth process. Part II. *Health Care Inf Age* 652–666
- Lloyd AM, Nightingale RW, Luck JF, Song Y, Fronheiser L, Cutcliffe H, Myers BS, Bass CRD (2015) The compressive stiffness of human pediatric heads. *J Biomech* 48:3766
- McPherson GK, Kriewall TJ (1980) The elastic modulus of fetal cranial bone: a first step towards an understanding of the biomechanics of fetal head molding. *J Biomech* 13(1):9
- Miller T (1997) A fast triangle–triangle intersection test. *J Graph Tools* 2:25
- Miller K, Joldes G, Lance D, Wittek A (2007) Total Lagrangian explicit dynamics finite element algorithm for computing soft tissue deformation. *Commun Numer Methods Eng* 23:121. <https://doi.org/10.1002/cnm.887>
- Mitsuhashi N, Fujieda K, Tamura T, Kawamoto S, Takagi T, Okubo K (2009) BodyParts3D: 3D structure database for anatomical concepts. *Nucl Acids Res* 37(Database):D782. <https://doi.org/10.1093/nar/gkn613>
- Moreau R, Pham MT, Silveira R, Redarce T, Brun X, Dupuis O (2007) Design of a new instrumented forceps: application to safe obstetrical forceps blade placement. *IEEE Trans Biomed Eng* 54(7):1280. <https://doi.org/10.1109/TBME.2006.889777>
- Morton G (1966) *A computer oriented geodetic data base and a new technique in file sequencing*. Tech. Rep. Ottawa, Ontario, Canada, IBM Ltd
- Nystrom R (2014) *Game programming patterns*. Genever Benning
- Oliveira DA, Parente MPL, Calvo B, Mascarenhas T, Jorge RMN (2017) The management of episiotomy technique and its effect

- on pelvic floor muscles during a malposition childbirth. *Comput Methods Biomech Biomed Eng* 20:11:1249
- Parente MPL, Jorge RMN, Mascarenhas T, Fernandes AA, Martins JAC (2009) The influence of the material properties on the biomechanical behavior of the pelvic floor muscles during vaginal delivery. *J Biomech* 42:1301
- Pietrzak G, Curnier A (1999) Large deformation frictional contact mechanics: continuum formulation and augmented Lagrangian treatment. *Comput Methods Appl Mech Eng* 177(3–4):351
- Silva M, Oliveira D, Roza T, Brando S, Parente M, Mascarenhas T, Jorge RN (2015) Study on the influence of the fetus head molding on the biomechanical behavior of the pelvic floor muscles, during vaginal delivery. *J Biomech* 48:1600
- Silva MET, Brando S, Parente MPL, Mascarenhas T, Jorge RMN (2017) Biomechanical properties of the pelvic floor muscles of continent and incontinent women using an inverse finite element analysis. *Comput Methods Biomech Biomed Eng* 0(0):1. <https://doi.org/10.1080/10255842.2017.1304542>
- Simo J, Laursen T (1992) An augmented Lagrangian treatment of contact problems involving friction. *Comput Struct* 42(1):97
- Sorbe B, Dahlgren S (1983) Some important factors in the molding of the fetal head during vaginal delivery—a photographic study. *Int J Gynecol Obstet* 21(3):205
- Taylor L (1989) PRONTO 3D: a three-dimensional transient solid dynamics program. U.S. Dept. of Commerce
- Taylor ZA, Cheng M, Ourselin S (2008) High-speed nonlinear finite element analysis for surgical simulation using graphics processing units. *IEEE Trans Med Imaging* 27(5):650. <https://doi.org/10.1109/TMI.2007.913112>
- Teran J, Blemker S, Hing V, Fedkiw R (2003) Finite volume methods for the simulation of skeletal muscle. *Proc ACM SIGGRAPH/Eurograph Symp Comput Anim M*:68
- Turnbull A (1957) Uterine contractions in normal and abnormal labour. *Am J Obstet Gynecol* 64(3):321
- US national library of medicine (1996) The visible human project. <https://www.nlm.nih.gov/research/visible/>
- Williams Obstetrics (2014) 24th edn, McGraw Hill, New York
- Yastrebov VA (2013) Numerical methods in contact mechanics. ISTE/Wiley, New York

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.